**Batch: D2 Roll No.: 16010221025**

**Experiment / assignment / tutorial No. 9**

**Grade: AA / AB / BB / BC / CC / CD /DD**

**Signature of the Staff In-charge with date**

|  |
| --- |
| **TITLE:**  Dynamic Memory Allocation. |

**AIM:** Program to demonstrate dynamic memory allocation using malloc() & free () function.

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Expected OUTCOME of Experiment:**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Books/ Journals/ Websites referred:**

1. Programming in C, second edition, Pradeep Dey and Manas Ghosh, Oxford University Press.
2. Programming in ANSI C, fifth edition, E Balagurusamy, Tata McGraw Hill.
3. Introduction to programming and problem solving , G. Michael Schneider ,Wiley India edition.
4. [**http://cse.iitkgp.ac.in/~rkumar/pds-vlab/**](http://cse.iitkgp.ac.in/~rkumar/pds-vlab/)

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Problem Definition:**

Implementing a C program to create a student list of a class using Dynamic memory allocation. It will have the details of students as roll number and name. Program should support the following operations (menu driven).

1. Insert

2. Delete

3. Display

use malloc for insert and free for delete

**Implementation details:**

#include <stdio.h>

#include <stdlib.h>

void main()

{

printf("Enter the number of players: ");

int num;

scanf("%d", &num);

char \*Name = (char \*)malloc((num+1)\*20\*sizeof(char));

int \*RollNo = (int\*)malloc((num+1)\*sizeof(int));

//data entry - after every input increment Name by 20 memory locations

// for rollno, increment 4 memory locations

int i=0;

printf("\nEnter the details of the players: ");

for(i = 0; i < num; i ++)

{

printf("\n-----\nPlayer %d \n", i+1);

printf("Name: ");

scanf("%s", Name+i\*20);

printf("Roll Number: ");

scanf("%d", RollNo+i\*sizeof(int));

}

printf("Data Stored...");

//insertion of data:

i++;

printf("Enter roll number and name of the student whose entry you want to insert: ");

printf("Name: ");

scanf("%s", Name+i\*20);

printf("Roll Number: ");

scanf("%d", RollNo+i\*sizeof(int));

//deletion of data:

char name[20];

printf("\nEnter the name of the student for his/her data needs to be deleted: \n");

scanf("%s", &name);

int flag = 0, j;

for(i = 0; i < num+1; i ++)

{

if(name == Name)

{

flag = i;

}

}

//whenever we need to display, we will skip the display of this element. So we don't delete but erase the path.

//display:

for(i = 0; i < num+1; i++)

{

if(i == flag)

{

continue;

}

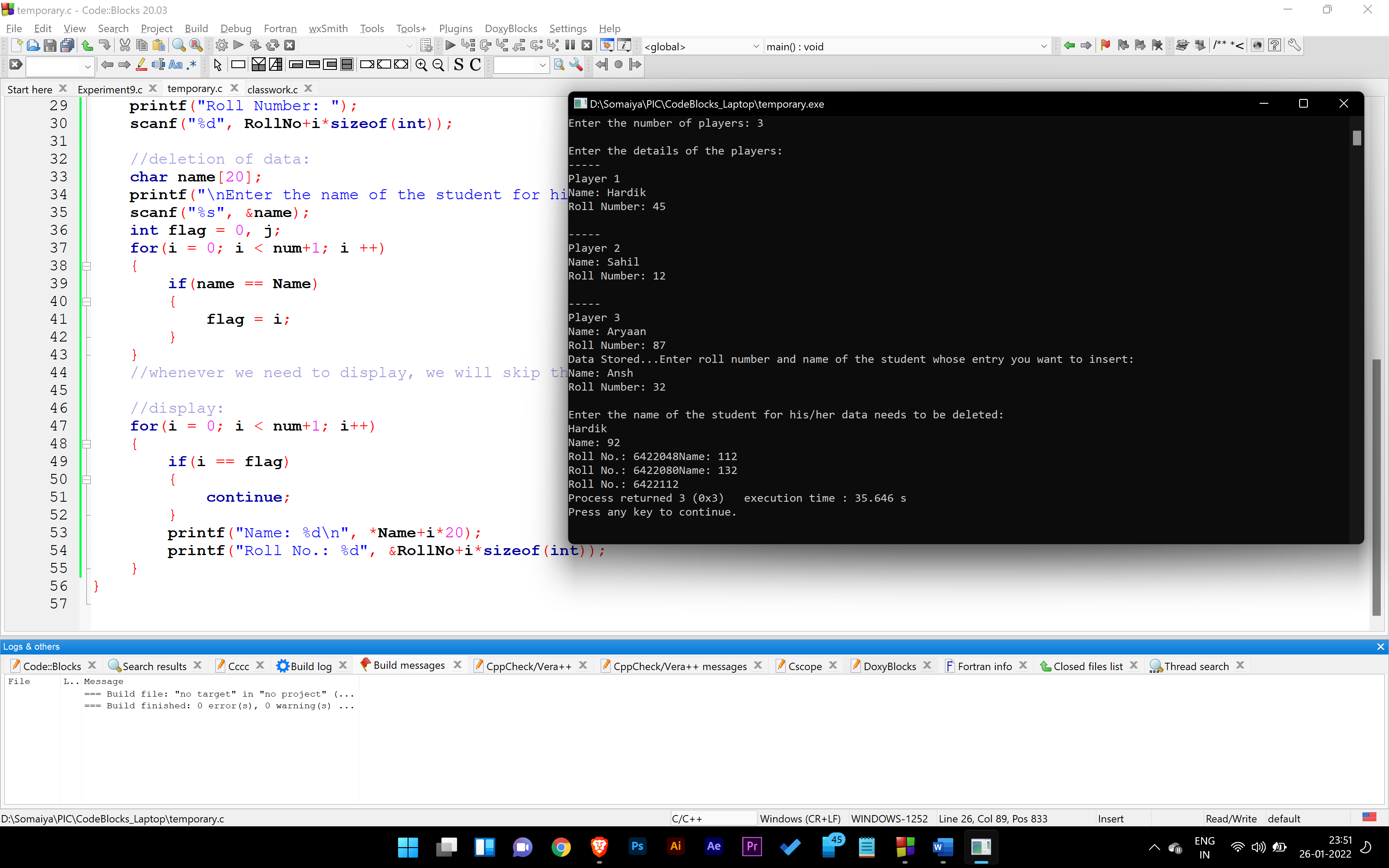
printf("Name: %d", \*Name+i\*20);

printf("Roll No.: %d", &RollNo+i\*sizeof(int));

}

}

**Output(s):**



**Conclusion:**

We learn how to code using malloc functions.

**Post Lab Descriptive Questions**

1. **What is the difference between malloc and calloc?**
2. **Consider the following C code. What will be the output?**

# include<stdio.h>

# include<stdlib.h>

void fun(int \*a)

{

  a = (int\*)malloc(sizeof(int));

}

int main()

{

  int \*p;

  fun(p);

  \*p = 6;

  printf("%d\n",\*p);

  return(0);

}

(A) Compiler Error

(B) 6

(C) Runtime Error

(D) Garbage Value

Ans.) (B.) 6

1. **Difference between Static and Dynamic Memory allocation**

Static memory is the kind of memory which is created while compile time. Dynamic memory is assigned during run time. Meaning the memory allocation happens while the program is running.

**Date: \_\_\_\_\_\_\_\_\_\_\_\_\_ Signature of faculty in-charge**